

## **Performance comparison of sender-based and receiver-based scheduling MAC protocols for underwater sensor networks**

Alfouzan, Faisal; Shahrabi, Alireza; Ghoreyshi, Seyed Mohammad; Boutaleb, Tuleen

*Published in:*

19th International Conference on Network-Based Information Systems (NBIS)

*DOI:*

[10.1109/NBiS.2016.3](https://doi.org/10.1109/NBiS.2016.3)

*Publication date:*

2016

*Document Version*

Author accepted manuscript

[Link to publication in ResearchOnline](#)

*Citation for published version (Harvard):*

Alfouzan, F, Shahrabi, A, Ghoreyshi, SM & Boutaleb, T 2016, Performance comparison of sender-based and receiver-based scheduling MAC protocols for underwater sensor networks, in *19th International Conference on Network-Based Information Systems (NBIS)*. IEEE, pp. 99-106, 19th International Conference on Network-Based Information Systems (NBIS), Ostrava, 7/09/16. <https://doi.org/10.1109/NBiS.2016.3>

### **General rights**

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

### **Take down policy**

If you believe that this document breaches copyright please view our takedown policy at <https://edshare.gcu.ac.uk/id/eprint/5179> for details of how to contact us.

# Performance Comparison of Sender-based and Receiver-based Scheduling MAC Protocols for Underwater Sensor Networks

Faisal Alfouzan, Alireza Shahrabi, Seyed Mohammad Ghoreyshi, Tuleen Boutaleb

School of Engineering and Built Environment, Glasgow Caledonian University, Glasgow, G4 0BA, United Kingdom

{Faisal.Alfouzan, A.Shahrabi, Seyed.MohammadGhoreyshi, T.boutaleb}@gcu.ac.uk

**Abstract**—Underwater Sensor Network (UWSN) is significantly different from terrestrial network in that acoustic signal is mainly used as the communication medium. The acoustic signal imposes high bit error rate, limited bandwidth and long propagation delay which, to some extent, influences the design of various services in UWSNs. MAC protocol, however, is one of the protocols which has been completely affected by the unique features of underwater environment. For UWSNs, energy-efficiency is now a major concern due to difficulties of replacing or recharging batteries. In order to save energy, most of the proposed MAC protocols follow a pattern of sleeping and listening mode, since the sleep mode energy consumption is much less than that of the idle listening mode. Hence, the energy efficiency is directly dependent upon how the sleeping and listening modes are scheduled at each node. In some protocols, sender-based scheduling, the schedule is specified by the sending nodes, whereas in receiver-based scheduling the receiving nodes decide to schedule data transmission. In this paper, we evaluate the UWAN-MAC, as a sender-based, and R-MAC, as a receiver-based scheduling. Our simulation results show that the receiver-based always performs well in terms of energy consumption for small size networks and under light traffic. However, by increasing the number of nodes or traffic load, the sender-based highly outperforms the receiver-based scheduling protocols in terms of end-to-end delay and throughput.

**Index Terms**—Underwater MAC protocols, Underwater acoustic sensor networks, Aqua-Sim simulator.

## I. INTRODUCTION

Over the last few years, there has been an increasing interest to develop and use Underwater Sensor Networks (UWSNs). The most important reason of that is to improve ocean exploration and support the demand for various civilian and military aquatic applications [1]. Since the earth's oceans almost cover 70 percent of its surface, resource discovery in underwater is considered as the main objective in order to reduce dependence on the land resources [2].

Similar to other types of sensor networks, an UWSN consists of a variable number of sensors that are deployed to perform collaborative monitoring tasks over a given area [3]. Those sensor nodes are used in applications such as environmental monitoring, oceanographic data collection, early warning systems, tactical surveillance, assisted navigation and exploration of oil-well [4]. However, UWSNs are significantly different from the terrestrial (radio) sensor networks since they mainly use sound waves as the communication medium. Hence, sensor nodes in water use acoustic channel as the communication method with 1500m/s speed, which is five orders of magnitude lower than that of radio signals. As a result of the lower propagation speed, higher propagation delays occur in communication, even between two neighbouring

nodes [5]. Implementing UWSNs are very costly since they usually must cover a very large area of the ocean environment, which occasionally leads to sparse deployment [6].

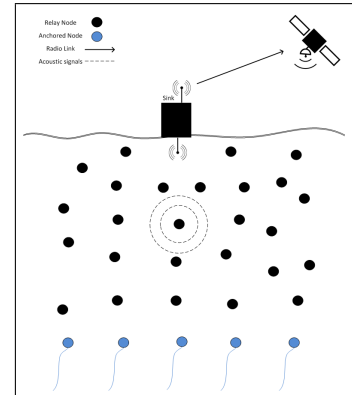


Fig. 1: Network architecture

Fig. 1 shows how underwater networks sensors are distributed and deployed in different places and depths of the ocean environment in order to collect information, and then forward them to the sink on the surface. Thereafter, collected information can be transmitted from the sink to the monitoring centre by using other wireless communication methods, such as satellite communication [7].

Using acoustic signal variously affects the design of different services in UWSNs [8]. However, it has completely changed the design of MAC protocols compared to that of terrestrial networks [9, 10]. Apart from long propagation delay, energy-efficiency is also a major concern due to difficulties of replacing or recharging batteries. In order to save energy, most of the proposed MAC protocols follow a pattern of sleeping and listening mode, since the sleep mode energy consumption is much less than that of the idle listening mode. Hence, the energy efficiency is directly dependent upon how the sleeping and listening modes are scheduled at each node. From this perspective, MAC protocols are classified in either sender-based or receiver-based scheduling protocols. The schedule is specified by the transmitting node in the former class in by the receiving node in the later class.

In this paper, we evaluate the performance of UWAN-MAC, as a sender-based, and R-MAC, as a receiver-based scheduling using Aqua-sim, a ns-2 based simulator for underwater acoustic networks. Different scenarios are created for two topologies, tree and star, to study their scalability and performance under light to heavy traffic.

The rest of this paper is organised as follows: In Section II, we review the related work. In Section III, the sender-based and the receiver-based scheduling protocols are presented. Section IV evaluates the performance of sender-based and receiver-based scheduling MAC protocols through simulations. Finally, Section V concludes the paper.

## II. RELATED WORK

The propagation delay in underwater environment significantly affects the design of MAC protocols for UWSNs. Using this fact, Liao *et al.* [6] have shown that the terrestrial MAC protocols can not operate reliably and efficiently in UWSN because they do not consider the long propagation delay of acoustic signals. Almost all energy-efficient UWSNs MAC protocols reported in the literature follow a contention-based approach, which means that node contend to access the shared medium with the possibilities of having collisions. In one group, handshaking, a sending and receiving nodes exchange some control packets before sending data packets trying to avoid any possible collision. Whereas in another group, random access, nodes attempt randomly to access the medium. When a data packet arrives at a receiver, if there is no packets coming from other nodes, the receiver can receive this packet successfully. The ALOHA is a random-access method which is not efficient enough to use due to high number of retransmissions and, therefore, energy consumption.

Most of the efforts in the design of the MAC protocol for UWSNs have focused on the handshaking group such as R-MAC, S-FAMA and DACAP. Each design of these MAC protocols has different requirements than another. Some variances based on the handshaking group have been proposed.

Slotted Floor Acquisition Multiple Access (*S-FAMA*) [11] protocol wastes the energy due to the idle listening and overhearing. It combines carrier sensing with a dialog between the sender and receiver before transmitting the data. In other words, *S-FAMA* uses request-to-send/ clear-to-send (RTS/CTS) control packets between the source and destination prior to the data transmission, and a packet (e.g. RTS, CTS, DATA and ACK) should be transmitted at the beginning of each slot. Although *S-FAMA* avoids DATA packet collisions without any requirement on the packet size, it requires clock synchronization between nodes which is difficult to achieve in UWSNs.

In [12], a Distance-Aware Collision Avoidance Protocol (*DACAP*) like *S-FAMA* is proposed which combines carrier sensing and an exchange of (RTS/CTS) control packets before transmitting data, but it does not require any synchronization between nodes. However, exchanging (RTS/CTS) control packets between the source and destination to eliminate data packet collisions, consumes a large amount of energy in underwater sensor networks because of the long propagation delay. In order to completely avoid collision among data packets, two conditions have to be satisfied: (1) the period of RTS should be greater than the maximum propagation delay; and (2) the period of CTS should be greater than that of RTS plus twice the maximum propagation delay plus the hardware transmit-to-receive transmission time [5]. Although, the RTS

packet and CTS packet exchange method consumes more energy, it can significantly mitigate the hidden and exposed terminal problems [6].

On the other hand, there are two methods of the random access in the classification of contention-based MAC protocols, which are ALOHA and Carrier Sense Multiple Access (CSMA). CSMA is multiple nodes share the transmission medium randomly without any control such as UWAN-MAC and T-Lohi protocols. Some variances based on the random access group have been proposed.

In [13], Syed *et al.* represent an idea to improve the system utilization by reducing the overhead, i.e. the number of transmitted control packets. This is proposed by the Tone-Lohi (*T-Lohi*) protocol where all the nodes send the short packets (tone packets) to inform their neighbours before sending any data. These tone packets can quickly make reservation in an energy-efficient way. Every node transmits its own tone, listening to the channel before sending data packets, and then each node counts how many other nodes do the same based on the number of tone packets received during the contention rounds (CRs).

Many studies, however, have focused on developing an efficient MAC protocol particularly for UWSN by adopting a schedule-based scheme, which determines an effective slot for the sensor node to transmit.

## III. SENDER-BASED AND RECEIVER-BASED SCHEDULING PROTOCOLS

Due to difficulty of replacing or recharging of sensor batteries, energy-efficiency is considered as a major factor when designing a new MAC protocol for UWSNs. In order to save energy, most of the proposed MAC protocols follow a pattern of sleeping and listening mode, since the sleep mode energy consumption is much less than that of the idle listening mode. Hence, the energy efficiency is directly dependent upon how the sleeping and listening modes are scheduled at each node. From this perspective, all reported contention-based MAC protocols can be divided into two classes; receiver-based and sender-based scheduling protocols. In receiver-based protocols, all nodes specify when they are in listening mode and able to receive data. Other nodes then need to adjust their transmissions in order to ensure that their data is arrived at the receiver side at the specified slot time. On the other hand, the sender-based protocols implies the receiving destination to adjust the listening slot time in a way to be able to receive the transmitted packets. Each class has its own advantages and disadvantages which can be discussed when comparing two protocols, one from each class. In this paper we study the performance of receiver-based and sender-based protocols by selecting two most cited protocols, one from each category. R-MAC and UWAN-MAC are two receiver-based and sender-based protocol, respectively, which follow different patterns of sleeping and listening modes in order to save energy.

Before discussing about our evaluation method and its results, we briefly explain each protocol and their main operational characteristics in this section.

### A. UWAN-MAC Protocol

UWAN-MAC protocol has been proposed by Park and Rodoplu [14] with the aim of increasing energy efficiency. Therefore, to reduce the energy consumption in every node, UWAN-MAC provides an appropriate method by increasing the sleep mode rather than the idle listening mode. This is based on the fact that sleep mode consumes less energy than the idle listening mode. In particular, UWAN-MAC is an energy efficient MAC protocol designed for underwater network. It belongs to a random access approach with no handshaking between senders and receivers. For instance, when the sender transmits a packet, it actually informs the receiver when it is supposed to send the next packet, and then the neighbours will overhear the packet to avoid the possible collisions [15]. The basic idea of UWAN-MAC protocol is shown in Fig. 2, which demonstrates how to achieve a locally synchronized schedule even if the propagation delay is unknown.

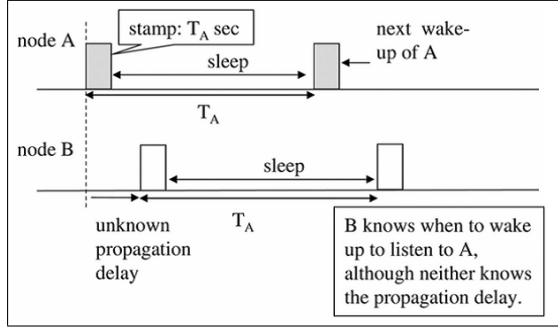


Fig. 2: The idea of the UWAN-MAC protocol [14]

In Fig. 2, node A broadcasts a SYNC packet which contains A's transmission cycle period " $T_A$ " at the beginning of its cycle period, and then goes to sleep. It is assumed that node B is located near to node A receives A's SYNC packet. This allows B to wake up exactly at the correct time of the next cycle to listen to node A without any knowledge of the propagation delay. Therefore, it is not necessary to provide clock synchronization, because nodes use the relative time stamps.

In the initialization period of this protocol, each node selects the initial transmission time randomly and then broadcasts its SYNC packet while at the same time it receives the neighbours' SYNC packets. Each SYNC packet includes  $T_i$  to inform all the neighbours that the new packet will be sent again after this period of time. When the initialization phase is completed, every node knows when it needs to wake up for listening and transmitting by following its established schedule.

Due to UWAN-MAC is designed to use only one control packet, it has been known as a bandwidth efficient protocol. On the other hand, a collision may occur in the following two ways:

- a "transmit-receive collision" may occur when a node transmits packet while another packet from a neighbour is arrived.

- a "receive-receive collision" may occur if more than two packets arrive at a node at the same time and the node cannot decode either of the packets.

### B. R-MAC Protocol

R-MAC protocol has been proposed by Xie and Cui [5]. The main design goals of R-MAC are energy efficiency and fairness. In this protocol, data packet collision is completely avoided by scheduling the transmissions of control packets and data packets. R-MAC not only avoids data packet collision, but it also can solve the exposed terminal problem inherited from the RTS/CTS-based protocols. In addition to solve the exposed terminal problem, it can save the energy and also support the fairness. However, instead of using RTS/CTS exchange control packets in R-MAC to avoid data packets collisions, the transmission of these packets is scheduled for both sender and receiver. Furthermore, in order to reduce the energy consumption on the idle state and overhearing, every node operates in the listen and sleep modes periodically.

In R-MAC, every node has three phases called *latency detection*, *period announcement* and *periodic operation*. Generally, the first two phases are employed to generate synchronization between nodes in the neighbourhood, and the last phase is for listening/sleeping operation.

In the latency detection phase, every node has to be power-on and randomly selects a time to broadcast a control packet, called *ND* (Neighbor Discovery). When a node receives *NDs* from its neighbours, it records the arrival times of these packets, and then randomly selects a time to transmit *ACK-ND* (acknowledge packet). *ND* and *ACK-ND* have exactly the same packet size. After receiving *ACK-ND*, a sender can compute the propagation latency. In this phase, the propagation latency can be calculated as follows:

$$L = \frac{I_1 - I_2}{2} \quad (1)$$

where  $I_2$  represents the duration from the arrival time of the ND packet to the transmission time of the ACK-ND, whereas;  $I_1$  indicates the interval from the time that the corresponding ND packet is transmitted to the arrival time of the ACK-ND. Then  $L$  is equal to the propagation latency.

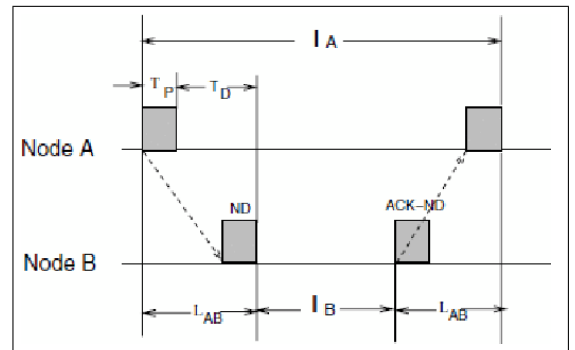


Fig. 3: Latency measurement [5]

For instance, Fig. 3 illustrates that node A sends a ND packet and records the time. Upon receiving this packet, node B randomly delays some period of times  $I_B$  and sends an ACK-ND packet back to the node A. Node B puts the time interval  $I_B$  in the ACK-ND packet, along with its ID and ND's packet ID. Upon receiving the ACK-ND packet, node A calculates the time interval from the transmission time of its ND packet to the interval time of ACK-ND packet,  $I_A$ . Then node A calculates the propagation latency to node B as follows:

$$L_{AB} = \frac{I_A - I_B}{2} \quad (2)$$

In the announcement phase period, each node randomly selects its own time of the listen/sleep periodic operations and broadcasts this time. After receiving the broadcast packets, called SYN (Synchronization packet), each node converts the received schedules to its own schedule. In this phase, each node records the schedules of its neighbours relative to its own schedule.

In the last phase, every node periodically wakes up and sleeps as one period called the listen/sleep cycle. In this periodic operation phase, nodes communicate by exchanging REV/ACK-REV/DATA/ACK-DATA messages. When a node has data to send, it first sends a REV (reservation message) to reserve a time slot at the receiver. The receiver will notify all its neighbours, when it is ready to receive data, by sending an ACK-REV message. All nodes receiving the ACK-REV packet will be silent in their corresponding time slots, except the sender node. Therefore, sender is able to send data packet at the reserved time slot. However, the first portion of this window is exclusively reserved for the ACK-REV messages. This reserved portion is called R-Window, which is a maximum duration of the control packet. When a node receives ACK-REV message in its R-Window, it knows the duration of the following data transmission and keeps silent in this period of time [16].

In R-MAC, the collisions can only be occurred in ACK-REV messages. Therefore, if the collisions occur in ACK-REV, it will back-off. Hence, only ACK-REV message can be transmitted to any node within its R-Window, whereas REV and ACK-DATA control packets are scheduled to arrive in the listen window, and DATA packets are scheduled to arrive at the receiver in the reserved time slot.

#### IV. PERFORMANCE EVALUATION

In this section, we present the performance results of UWAN-MAC and R-MAC protocols by using Aqua-Sim which is an NS-2 based simulator for UWSNs [17]. The following simulation parameters are used in all scenarios. The power consumption on transmission mode is 2 Watts; the power consumption on receiver mode is 0.75 Watts; and the power consumption on sleep mode is 8 mW. The control packet size is 5 Bytes and the data packet size is 60 Bytes. The bit rate is 10 Kbps and the maximum transmission range is 90 meters. The routing protocol is Vector Based Forwarding

(VBF) and the bandwidth is 90 Kbps. We set the simulation time to three hours for the first scenario and one hour for the second scenario. This simulation shows how the two protocols perform in real underwater networks. In our simulation, we consider two topologies: Star and Tree. In the single-hop star topology, all nodes in the network are directly connected to each other and packets are sent to a single receiving node within an area of  $100m \times 100m \times 100m$ . In the multiple-hop tree topology, however; all nodes are distributed in a  $100m \times 100m \times 200m$  area. The sink node is placed at the top of the simulation area.

##### A. Performance Metrics

We define four metrics to quantify the performance of UWAN-MAC and R-MAC: *end-to end delay*, *throughput*, *energy consumption* and the *packet drop rate*.

End-to-end delay is the average time interval from the source to the destination for each successfully delivered packet. Throughput is defined as the number of successful received packets divided by the simulation time. Energy consumption is obtained by dividing the overall energy consumption in the network by the successfully delivered data packets, which is measured in joules per packet. Packet drop rate is defined as the number of loss packets divided by the simulation time.

##### B. Single-hop Star Topology

We evaluate the performance of UWAN-MAC and R-MAC protocols by using a star topology, as shown in Fig 5. The topology length, width and depth are  $100m \times 100m \times 100m$ , respectively. In this network, node 0 at the middle acts as a sink and all other nodes directly transmit the packets to the sink. All the anchored nodes can hear each other. However, there are two scenarios used in the single-hop star topology. Firstly, we measure the end-to-end delay, throughput, energy consumption and packet drop rate as a function of the traffic rate. In this scenario, the traffic rate varies from 0.02 to 0.40 packets per second with only one sink and four anchored nodes. Secondly, we measure the end-to-end delay, throughput, energy consumption and packet drop rate as a function of number of nodes in order to study each MAC protocol scalability. The numbers of nodes are 3, 6, 9, 12, 15 and 18; where only one sink is located in the middle of the network and all the other nodes send packets to the sink. To do this, we fix the traffic rate at 0.12 packets per second. In addition, to have a fair comparison, we set the same parameters for UWAN-MAC and R-MAC.

Fig. 4 (a) displays end-to-end delay as a function of traffic rate. The graph shows a sharp increase with R-MAC protocol from 0.05 to 0.15 data rates by reaching approximately 0.8 second per packet as end-to-end delay. Nonetheless, the delay will remain constant when R-MAC reaches its maximum level, because R-MAC simply drops the extra packets in the high traffic rate. This is due to the fact that, in high traffic rate, receiver is not flexible to accept more packets in its scheduling. However, the delay in UWAN-MAC gradually increases to

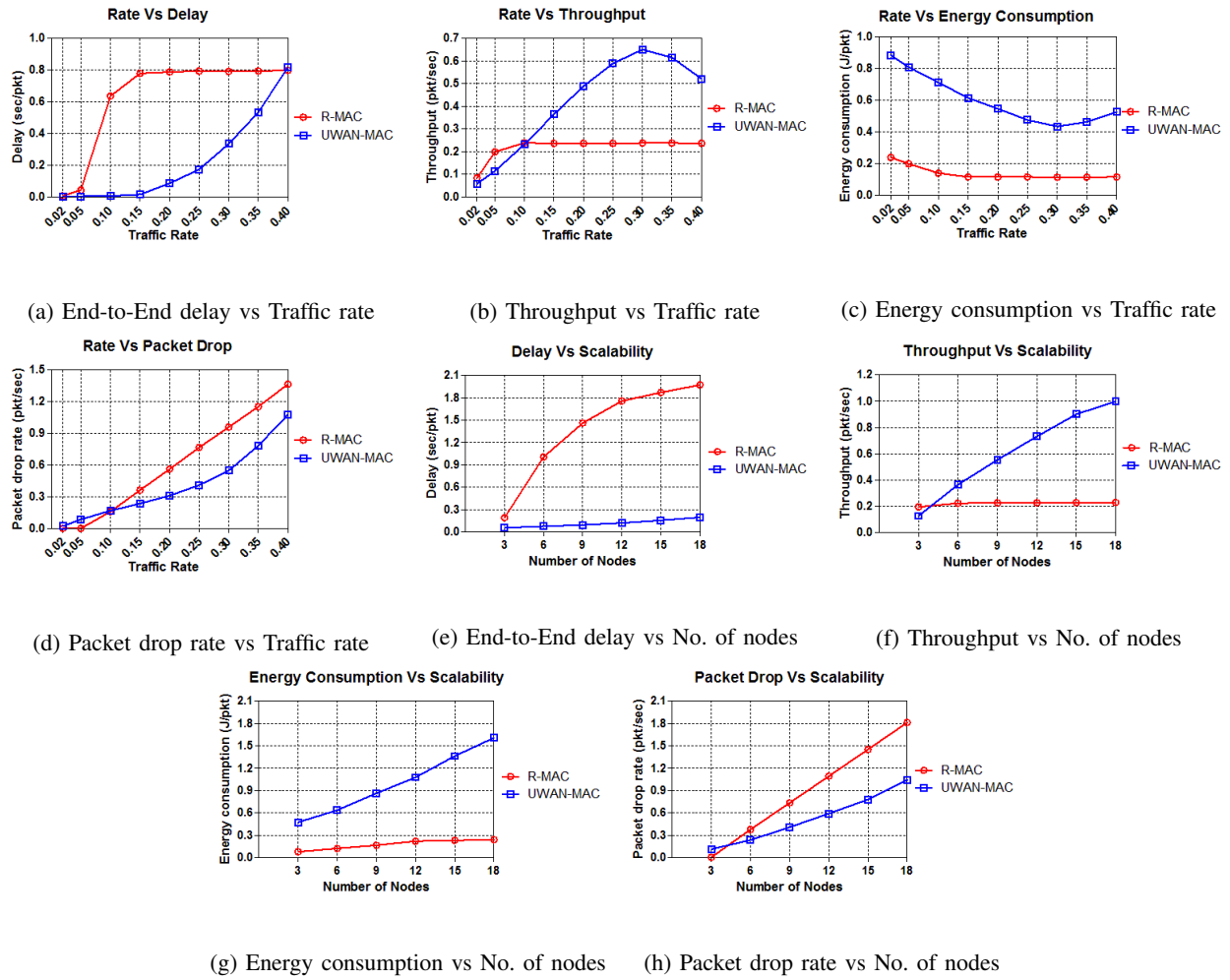


Fig. 4: Single-hop network: end-to-end delay, throughput, energy consumption and packet drop rate vs Traffic rate and Number of nodes

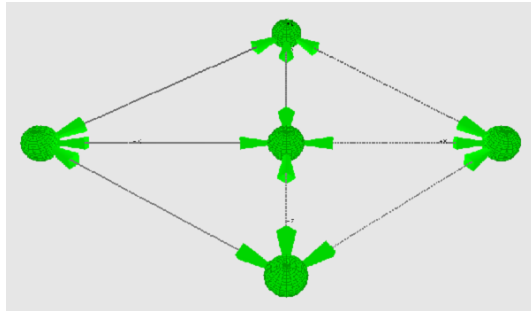


Fig. 5: Sensor nodes in the star topology

reach 0.8 second per packet. It should be mentioned that UWAN-MAC is more flexible to accept more packets. Thus, accepting more packets logically leads to more delay in the scheduling of the accepted packets. On the other hand, Fig. 4 (e) shows the delay versus number of nodes. In R-MAC with this scenario, the delay does not reach to any maximum value with increasing the number of nodes. This is due to the fact that data generated rate is fixed in an acceptable value for R-MAC which it can work properly, but increase

in the number of nodes only increases the delay because of the topology constraints. In UWAN-MAN, the delay slightly increases because of its scalability feature.

As can be seen in this scenario, the delay difference between R-MAC and UWAN-MAC is enlarged. This is mainly due to the randomness transmitting of UWAN-MAC protocol, while R-MAC trades delay for energy efficiency and fairness. Moreover, R-MAC is a receiver-based scheduling which means that the receiver must transmit an ACK-REV to each node of its neighbours. As a result of this, when the number of nodes increases, the end-to-end delay of R-MAC is significantly increased.

As can be seen from Fig. 4 (b), the network throughputs of both UWAN-MAC and R-MAC are drawn as a function of traffic rate. The throughput in UWAN-MAC sharply increases at the first (from 0.02 until 0.3) and then decreases. This is mainly because of the more intensive channel competition as well as the relatively increased hidden terminal problems. In low traffic rates; however, R-MAC receives a large amount of packets until 0.1 and then remains constant to be just over 0.2 packet per second. This is because the traffic rate is exceeded



the channel capacity. Hence, only some data packets are sent and other packets are dropped. On the other hand, UWAN-MAC can handle the lower amount of packets than R-MAC in low traffic rate. Therefore, R-MAC outperforms UWAN-MAC in a very low traffic because a receiver-based approach can have better scheduling than sender-based (randomness). Nevertheless, Fig. 4 (f) shows the throughput versus the number of nodes. In R-MAC, when the number of nodes increases with a fixed data rate, it reaches a saturated point because it is not a scalable approach. In UWAN-MAC, when the number of nodes increases, the throughput greatly increases to reach 1 packet per second with 18 nodes. This is because it benefits from a high scalable scheduling to handle more packets.

Comparing Fig. 4 (c) and (g), we observe that UWAN-MAC consumes more energy per packet in both traffic rate and number of nodes than R-MAC. In fact, when the traffic rate varies from 0.02 to 0.3, the energy consumption of UWAN-MAC significantly decreases, from 0.9 to 0.4 joules per packet. This is because UWAN-MAC can handle more packets with a little increase in the total energy consumption. Thus, energy consumption in terms of the energy per packet will be reduced in this protocol. UWAN-MAC also uses a random access approach which does not have a handshaking. After traffic rate 0.3, the energy consumption increases slightly to reach 0.5 joules per packet, due to more intensive channel competition and increase in the hidden terminal nodes. On the other hand, when the number of nodes increases, the energy consumption of UWAN-MAC considerably increases by reaching almost 1.6 joules per packet with 18 nodes. Whereas, the energy consumption of R-MAC increases slightly to achieve a constant level just under 0.3 joules per packet. This is mainly because that R-MAC strictly schedules the transmission of packets to completely avoid data packets collisions, while; UWAN-MAC has no handshaking process and cannot reduce the collisions which caused by the hidden terminal problems.

Fig. 4 (d) shows the packet drop rate as a function of traffic rate. The packet drop rate of R-MAC is very low at 0.02 and 0.05 data rates, and then sharply increases to achieve almost 1.3 packet per second at 0.4 data rate. This is because R-MAC is controlled by the receiver node and when the traffic rate increases, node cannot reserve the channel immediately, resulting in high drop rate and therefore low data throughput. However, the packet drop rate of UWAN-MAC is very low at the low traffic rate with almost 0.1 packet per second at 0.05 traffic rate, and then gradually increases to drop approximately 0.5 packet per second at 0.3 traffic rate. Thereafter, the packet drop rate sharply increases to 1.1 packet per second at 0.4 traffic rate. Overall, the packet drop rate of UWAN-MAC slightly increases at the first and then rapidly increases with higher traffic rates. The main reason for this phenomenon is that the packet scheduling is controlled by the sender node which may cause more intensive channel competition between nodes, and also because there is no handshaking in UWAN-MAC, collisions are more likely to be occurred by the hidden terminal problem. The packet drop rate versus the number of nodes is shown in Fig. 4 (h). With three nodes, R-

MAC drops very small amount of packets than UWAN-MAC. However, by increasing the number of nodes, UWAN-MAC drops lower packets than R-MAC. Therefore, UWAN-MAC is more reliable and suitable with a large number of nodes, in contrast to R-MAC. This is mainly because that the R-MAC traffic rate is exceeded the channel capacity, which only allows some packets to send and the rest of packets are dropped. The main reason for low drop rate in UWAN-MAC is that it is more flexible to accept more packets and it is also a randomness protocol. Thus, UWAN-MAC is highly scalable for a larger network size.

### C. Multiple-hop Tree Topology

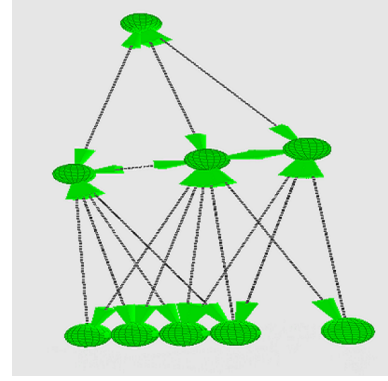


Fig. 6: Sensor nodes in the tree topology

In this set of simulations, we investigate the performance of UWAN-MAC and R-MAC protocols by using multi-hop tree topology network as illustrated in Fig. 6. The topology length, width and depth are  $100m \times 100m \times 200m$ , respectively. As can be seen, there are three layers located at different depths in order to reach the sink node at the top layer. Therefore, the bottom layer of the network contains sender nodes transmitting data to the relay nodes in the middle layer. This layer also contains the relay nodes which forward the packets from the bottom layer to the upper layer. In this scenario, there are two functions used in the multi-hop tree topology. First, we evaluate the end-to-end delay, throughput, energy consumption and packet drop rate as a function of traffic rates. The traffic rate varies from 0.02 to 0.40 packets per second with only one sink, three relay nodes and five anchored nodes. Second, we measure the end-to-end delay, throughput, energy consumption and packet drop rate as a function of number of nodes. The numbers of nodes are 3, 6, 9, 12, 15 and 18. In this function, we fix the traffic rate to be 0.12 packets per second, as widely used in the literature.

Fig. 7 (a) depicts the end-to-end delay as a function of the traffic rate. The graph shows that there is a rapid increase with R-MAC protocol by reaching approximately 0.7 packet per second as end-to-end delay. When R-MAC reaches its maximum level, the delay will remain constant because R-MAC simply drops the extra packets in the high traffic rate. The receiver in R-MAC is not flexible enough to accept more packets in its scheduling. On the other hand, the delay

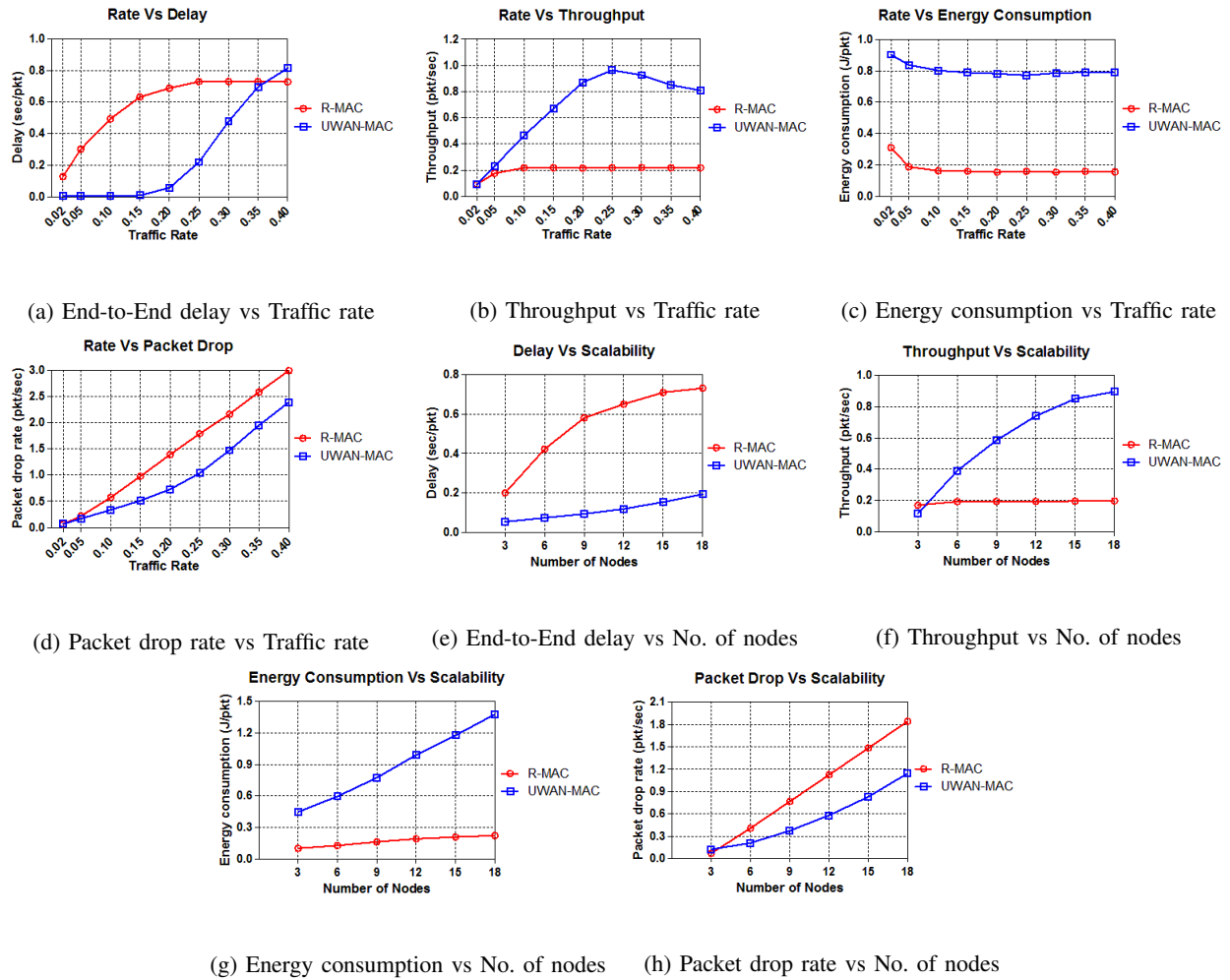


Fig. 7: Multiple-hop network: end-to-end delay, throughput, energy consumption and packet drop rate vs Traffic rate and Number of nodes

in UWAN-MAC gradually increases to reach 0.8 packet per second. This is due to the fact that UWAN-MAC is more flexible to accept more packets. Thus, accepting more packets logically leads to more delay in the scheduling of the accepted packets. The end-to-end delay versus the number of nodes is shown in Fig. 7 (e). We can observe that the end-to-end delay in R-MAC significantly increases by increasing the number of nodes until it reaches its maximum value, because it exceeds the channel capacity. Another reason is that the receiver in R-MAC must transmit an ACK-REV to each node of its neighbours. However, in UWAN-MAN, the delay slightly increases because of its scalability features.

As can be seen from Fig. 7 (b), the network throughputs of both UWAN-MAC and R-MAC are drawn as a function of traffic rate. This allows us to make several observations between UWAN-MAC and R-MAC. Firstly, the throughput of R-MAC is 0.1 packet per second at 0.02 data rate, and then increases to achieve almost 0.2 packet per second at 0.05 data rate. Thereafter, the throughput of R-MAC is fixed at 0.22 packet per second. This is mainly because that the traffic rate is exceeded the channel capacity. Therefore, only some data

packets are accepted and some of them are dropped, in a way that the traffic rate has no effect on the throughput any more. Secondly, the throughput of UWAN-MAC increases at the first (from 0.02 until 0.25) and then decreases. The reduction in the throughput is mainly because of the more intensive channel competition. Another reason is that UWAN-MAC does not perform handshaking and consequently collisions may be caused by the hidden terminal problems. Thus, R-MAC is better than UWAN-MAC in a very low traffic because a receiver-based approach can have better scheduling than sender-based (randomness). Fig. 7 (f) shows the throughput versus number of nodes. In R-MAC, when the number of nodes increases with a fixed data generated rate, it reaches saturated point because it is not a scalable approach. However, in UWAN-MAC, when the number of nodes increases, the throughput shows high increase by achieving 0.9 packet per second with 18 nodes. This is because it can handle more packets and highly scalable schedule.

Fig. 7 (c) shows the energy consumption as a function of the traffic rate. We can observe that UWAN-MAC consumes much more energy than R-MAC in all data rates. In particular,



the energy consumption of UWAN-MAC slightly decreases from 0.9 joules per packet to stay in 0.8 joules per packet. However, R-MAC has a very low energy consumption which significantly decreases from 0.3 joules per packet to consume just under 0.2 joules per packet. In the literature [5], it has been argued that R-MAC trades end-to-end delay for energy efficiency and fairness. Fig. 7 (g) demonstrates the energy consumption as a function of the number of nodes. We can observe that when the number of nodes increases, the energy consumption of UWAN-MAC considerably increases by reaching 1.3 joules per packet with 18 nodes. This is because that UWAN-MAC does not perform handshaking process and consequently increases the collisions caused by the hidden terminal problems. On the other hand, the energy consumption of R-MAC increases slightly to achieve a constant level just under 0.3 joules per packet from 3 to 18 nodes. This is mainly because that R-MAC strictly schedule the transmission of control and data packets by sacrificing the end-to-end delay for more energy efficiency and fairness.

Fig. 7 (d) depicts the packet drop rate vs the traffic rate. Between 0.02 and 0.05 data rates, both UWAN-MAC and R-MAC drop the same amount of packets. We can observe that UWAN-MAC dropped lower packets than R-MAC from 0.1 to 0.4 data rates. This is due to that R-MAC is controlled by the receiver node and when the traffic rate increases, node cannot reserve the channel immediately, which causes high packet drop rate. The reason why UWAN-MAC drops less packets than R-MAC is that UWAN-MAC is a sender-based scheduling.

Fig. 7 (h) illustrates the packet drop rate as a function of the number of nodes. We can observe that when the number of nodes increases, UWAN-MAC drops the packets less than R-MAC. The packet drop rate of R-MAC sharply increases to drop almost 1.8 packet per second with 18 nodes. This is because that the receiver in R-MAC has to transmit an ACK-REV packet to every node of its neighbours which causes high drop rate. Whereas UWAN-MAC slightly increase to drop roughly 1.2 packet per second with 18 nodes. This is because that UWAN-MAC is more flexible to accept more packets and it is a randomness protocol. According to this result, it can be argued that UWAN-MAC is more scalable protocol.

## V. CONCLUSION

A comparative performance study of two different scheduling disciplines, UWAN-MAC as a sender-based and R-MAC as a receiver-based protocols, for underwater sensor networks has been presented in this paper. The main difference between these two classes of MAC protocols is that the receiver-based scheduling measures the propagation delay and accurately schedules the transmissions of control and data packets between nodes in order to completely avoid collisions, whereas the sender-based scheduling involves unknown propagation delays.

We have investigated several scenarios that are typical for underwater channel access studies; a single-hop and a multi-hop topology with different traffic rates and number of nodes

by using ns-2. Although each protocol has its own advantages and disadvantages in different scenarios. Our results show that R-MAC always performs well under light traffic for small networks in terms of energy consumption. However, by increasing the number of nodes, UWAN-MAC highly outperforms R-MAC in terms of end-to-end delay and throughput in both single-hop and multi-hop scenarios. UWAN-MAC still keeps a good throughput by increasing the traffic rate as well although it has a high end-to-end delay under high traffic rate. However, the main disadvantage of UWAN-MAC protocol is that collisions can still occur when a node is transmitting or receiving packets.

## REFERENCES

- [1] K. Chen, M. Ma, E. Cheng, F. Yuan, and W. Su, "A survey on mac protocols for underwater wireless sensor networks," *Communications Surveys & Tutorials, IEEE*, vol. 16, no. 3, pp. 1433–1447, 2014.
- [2] H. Yu, N. Yao, and J. Liu, "An adaptive routing protocol in underwater sparse acoustic sensor networks," *Ad Hoc Networks*, vol. 34, pp. 121–143, 2015.
- [3] C.-C. Hsu, K.-F. Lai, C.-F. Chou, and K. C.-J. Lin, "St-mac: Spatial-temporal mac scheduling for underwater sensor networks," in *INFOCOM 2009, IEEE*, pp. 1827–1835, 2009.
- [4] S. Xiong, C. Yuan, L. Tian, and Y. Zhan, "Ret-mac: A new fair mac protocol for underwater acoustic sensor network," *Journal of Distributed Sensor Networks*, 2013.
- [5] P. Xie and J.-H. Cui, "R-mac: An energy-efficient mac protocol for underwater sensor networks," in *Wireless Algorithms, Systems and Applications, 2007. WASA 2007. International Conference on*, pp. 187–198, IEEE, 2007.
- [6] W.-H. Liao and C.-C. Huang, "Sf-mac: A spatially fair mac protocol for underwater acoustic sensor networks," *Sensors Journal, IEEE*, vol. 12, no. 6, pp. 1686–1694, 2012.
- [7] S. M. Ghoreyshi, A. Shahrahi, and T. Boutaleb, "A novel cooperative opportunistic routing scheme for underwater sensor networks," *Sensors*, vol. 16, no. 3, p. 297, 2016.
- [8] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad hoc networks*, vol. 3, no. 3, pp. 257–279, 2005.
- [9] C.-C. Hsu, M.-S. Kuo, C.-F. Chou, and K. C.-J. Lin, "The elimination of spatial-temporal uncertainty in underwater sensor networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 21, no. 4, pp. 1229–1242, 2013.
- [10] Z. Li, Z. Guo, F. Hong, and L. Hong, "E 2 dts: an energy efficiency distributed time synchronization algorithm for underwater acoustic mobile sensor networks," *Ad Hoc Networks*, vol. 11, no. 4, pp. 1372–1380, 2013.
- [11] M. Molins and M. Stojanovic, "Slotted fama: a mac protocol for underwater acoustic networks," in *OCEANS 2006-Asia Pacific*, pp. 1–7, IEEE, 2007.
- [12] B. Peleato and M. Stojanovic, "Distance aware collision avoidance protocol for ad-hoc underwater acoustic sensor networks," *Communications Letters, IEEE*, vol. 11, no. 12, pp. 1025–1027, 2007.
- [13] A. Syed, W. Ye, J. Heidemann, et al., "T-lohi: A new class of mac protocols for underwater acoustic sensor networks," in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, 2008.
- [14] M. K. Park and V. Rodoplu, "Uwan-mac: An energy-efficient mac protocol for underwater acoustic wireless sensor networks," *Oceanic Engineering, IEEE Journal of*, vol. 32, no. 3, pp. 710–720, 2007.
- [15] Z. Peng, Y. Zhu, Z. Zhou, Z. Guo, and J.-H. Cui, "Cope-mac: a contention-based medium access control protocol with parallel reservation for underwater acoustic networks," in *OCEANS 2010 IEEE-Sydney*, pp. 1–10, 2010.
- [16] S. A. Samad, S. Shenoy, and G. S. Kumar, "Improving energy efficiency of underwater acoustic sensor networks using transmission power control: A cross-layer approach," in *Advances in Computing and Communications*, pp. 93–101, Springer, 2011.
- [17] P. Xie, Z. Zhou, Z. Peng, H. Yan, T. Hu, J.-H. Cui, Z. Shi, Y. Fei, and S. Zhou, "Aqua-sim: an ns-2 based simulator for underwater sensor networks," in *OCEANS 2009, MTS/IEEE Biloxi-Marine Technology for Our Future: Global and Local Challenges*, pp. 1–7, IEEE, 2009.